What is claimed is:

1. A state machine, the state machine being arranged to be used within a computing system that supports an enterprise platform, the state machine implementation comprising:

an entity bean class;

a home interface associated with the entity bean class, the home interface being arranged to create, find, and remove entity objects;

a remote interface associated with the entity bean class, the remote interface being arranged to drive the state machine in response to the input events; and

an entity object associated with the entity bean class, the entity object being arranged to represent an individual state machine.

2. A state machine according to claim 1 wherein the entity bean class, the entity object, the home interface, and the remote interface are parts of an entity bean.

3. A state machine according to claim 2 wherein the entity bean is an enterprise bean

4. A state machine according to claim 3 wherein the enterprise bean is an Enterprise JavaBean.

5. A state machine according to claim 2 wherein the entity bean is arranged to be deployed in a bean container that that is arranged to invoke instances of the entity bean class in response to invocations to the methods of the remote interface.

6. A state machine according to claim 5 wherein the bean container is further arranged to implement a timeout.

7. A state machine according to claim 5 wherein the entity bean uses container-managed persistence to maintain a state associated with the state machine.

8. A state machine according to claim 5 wherein the entity bean uses bean-managed persistence to main a state associated with the state machine.

9. A state machine according to claim 5 wherein the bean container is arranged to receive input events to the entity bean and to dispatch the received input events to the methods of the remote interface

10. A state machine according to claim 1 wherein the remote interface is arranged to define a method for an input event to which the state machine responds.

11. A state machine method used within a computing system that supports an enterprise platform comprising:

providing an entity bean class;

providing a remote interface associated with the entity bean class, the remote interface being arranged to drive the state machine in response to the input events;

providing an entity object associated with the entity bean class, the entity object being arranged to represent an individual state machine; and

providing a home interface associated with the entity bean class, the home interface being arranged to create, find, and remove entity objects.

12. A state machine method according to claim 11 wherein the entity bean class, the entity object, the home interface, and the remote interface are parts of an entity bean.

13. A state machine method according to claim 12 wherein the entity bean is an enterprise bean

14. A state machine method according to claim 13 wherein the enterprise bean is an Enterprise JavaBean.

15. A state machine method according to claim 12 wherein the entity bean is arranged to be deployed in a bean container that that is arranged to invoke instances of the entity bean class in response to invocations to the methods of the remote interface.

16. A state machine method according to claim 15 wherein the bean container is further arranged to implement a timeout.

17. A state machine method according to claim 15 wherein the entity bean uses container-managed persistence to maintain a state associated with the state machine.

18. A state machine method according to claim 15 wherein the entity bean uses bean-managed persistence to main a state associated with the state machine.

19. A state machine method according to claim 15 wherein the bean container is arranged to receive input events to the entity bean and to dispatch the received input events to the methods of the remote interface

20. A state machine method according to claim 11 wherein the remote interface is arranged to define a method for an input event to which the state machine responds.